

MONITORING IN TIME AND RELATIVE DIMENSIONS IN SPACE

Monitoring in multiple dimensions

Sven Putteneers

Saturday 2020-03-07

 **BUSINESS**



AGENDA

- 1 Our setup
- 2 Problem statement
- 3 Initial approach
- 4 Current solution
- 5 Example alert action
- 6 Future developments
- 7 Conclusion

0

INTRODUCTION

Who am I, who are we?

INTRODUCTION

\$ whoami

- Sven Putteneers
- 20+ years of Linux experience
- Monitoring has been part of my job for 3.5 years
- I ♥ Zabbix

INTRODUCTION

Who are we

- Now
 - Telenet BVBA
 - One of the largest telecom operators in Belgium
 - Somewhat recently acquired Nextel NV
- Past
 - Nextel NV
 - Telecom integrator
 - Lots of equipment on customer premises
 - Monitoring as a service
 - Multiple teams

1

OUR SETUP

OUR SETUP

- Zabbix LTS (we're currently on 4.0)
- ~1700 VPS
- We offer monitoring as a service
 - Heavy use of proxies
 - Different kinds of equipment to monitor and support
 - Multiple alarm regimes
 - 24x7
 - 8x5
 - No active alerts (useful for troubleshooting issues)
- Zabbix is the standby paging service

2

PROBLEM STATEMENT

Multi-tenancy, multiple teams, multiple alarm regimes... Ai ai ai!

AN INSPIRATIONAL QUOTE

- Monitoring is the art of getting the right information in front of the right people at the right time.
– *Source unknown*

PROBLEM STATEMENT

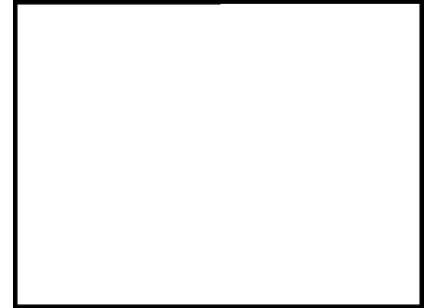
Monitoring in multiple dimensions

- Who must receive an alert?
 - Each technology is supported by a different team
- When to send alerts
 - Not all customers want 24×7 support
 - We don't want to wake on-call people when it's not needed
- Which customer does a device belong to?
- Automatic incident ticket creation is sometimes required
- ... and we want to keep this maintainable

PROBLEM STATEMENT

Monitoring in multiple dimensions

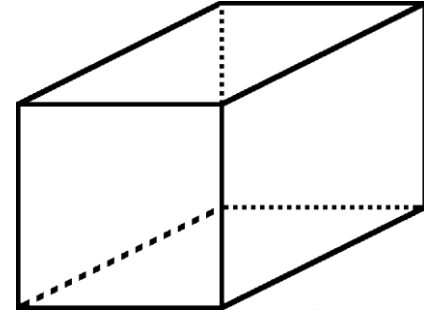
- Who must receive an alert?
 - Each technology is supported by a different team
- When to send alerts
 - Not all customers want 24×7 support
 - We don't want to wake on-call people when it's not needed
- Which customer does a device belong to?
- Automatic incident ticket creation is sometimes required
- ... and we want to keep this maintainable



PROBLEM STATEMENT

Monitoring in multiple dimensions

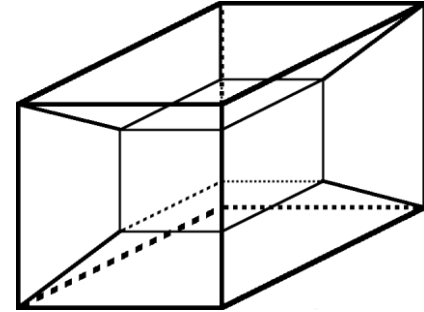
- Who must receive an alert?
 - Each technology is supported by a different team
- When to send alerts
 - Not all customers want 24×7 support
 - We don't want to wake on-call people when it's not needed
- Which customer does a device belong to?
- Automatic incident ticket creation is sometimes required
- ... and we want to keep this maintainable



PROBLEM STATEMENT

Monitoring in multiple dimensions

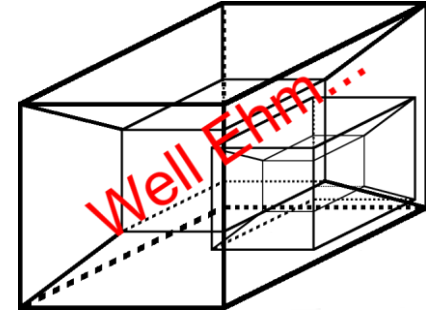
- Who must receive an alert?
 - Each technology is supported by a different team
- When to send alerts
 - Not all customers want 24×7 support
 - We don't want to wake on-call people when it's not needed
- Which customer does a device belong to?
- Automatic incident ticket creation is sometimes required
- ... and we want to keep this maintainable



PROBLEM STATEMENT

Monitoring in multiple dimensions

- Who must receive an alert?
 - Each technology is supported by a different team
- When to send alerts
 - Not all customers want 24×7 support
 - We don't want to wake on-call people when it's not needed
- Which customer does a device belong to?
- Automatic incident ticket creation is sometimes required
- ... and we want to keep this maintainable



3

INITIAL APPROACH

First implementation and its downsides

INITIAL APPROACH

First implementation

- Who
 - Alarms are sent to the team that is responsible for “most” devices
- When
 - Property of the customer
 - **All** devices for a given customer follow the same regime
- Which customer does a device belong to?
 - Customer name is part of the (Zabbix) hostname of the device
- All alarms of severity \geq High generate a support ticket

INITIAL APPROACH

First implementation

- Who
 - Alarms are sent to the team that is responsible for “most” devices
- When
 - Property of the customer
 - **All** devices for a given customer follow the same regime
- Which customer does a device belong to?
 - Customer name is part of the (Zabbix) hostname of the device
- All alarms of severity \geq High generate a support ticket

INITIAL APPROACH

First implementation

- Who
 - Alarms are sent to the team that is responsible for “most” devices
- When
 - Property of the customer
 - **All** devices for a given customer follow the same regime
- Which customer does a device belong to?
 - Customer name is part of the (Zabbix) hostname of the device
- All alarms of severity \geq High generate a support ticket

INITIAL APPROACH

First implementation

- Who
 - Alarms are sent to the team that is responsible for “most” devices
- When
 - Property of the customer
 - **All** devices for a given customer follow the same regime
- Which customer does a device belong to?
 - Customer name is part of the (Zabbix) hostname of the device
- All alarms of severity \geq High generate a support ticket

INITIAL APPROACH

Not bad, but not perfect either

- Who
 - “Mixed” contracts: devices of different teams at the same customer
 - Example: a customer with network equipment, but also security cameras
 - Alerts go to the wrong team from time to time
 - It’s not nice waking up from a pager call about a device you can’t do anything about
 - Figuring out who can actually fix the problem takes extra time... Not good for SLAs
 - The dominant type of device a customer has might change over time
- When
 - All devices generate alarms according to the same regime
 - It’s not possible to monitor some devices for data collection only
- Customer name as a prefix to the hostname
 - Requires discipline to name all devices according to the naming convention
 - Long customer names causes readability to suffer
 - “Creative” abbrevs. for customer names

INITIAL APPROACH

Not bad, but not perfect either

- Who
 - “Mixed” contracts: devices of different teams at the same customer
 - Example: a customer with network equipment, but also security cameras
 - Alerts go to the wrong team from time to time
 - It’s not nice waking up from a pager call about a device you can’t do anything about
 - Figuring out who can actually fix the problem takes extra time... Not good for SLAs
 - The dominant type of device a customer has might change over time
- When
 - All devices generate alarms according to the same regime
 - It’s not possible to monitor some devices for data collection only
- Customer name as a prefix to the hostname
 - Requires discipline to name all devices according to the naming convention
 - Long customer names causes readability to suffer
 - “Creative” abbrevs. for customer names

INITIAL APPROACH

Not bad, but not perfect either

- Who
 - “Mixed” contracts: devices of different teams at the same customer
 - Example: a customer with network equipment, but also security cameras
 - Alerts go to the wrong team from time to time
 - It’s not nice waking up from a pager call about a device you can’t do anything about
 - Figuring out who can actually fix the problem takes extra time... Not good for SLAs
 - The dominant type of device a customer has might change over time
- When
 - All devices generate alarms according to the same regime
 - It’s not possible to monitor some devices for data collection only
- Customer name as a prefix to the hostname
 - Requires discipline to name all devices according to the naming convention
 - Long customer names causes readability to suffer
 - “Creative” abbrevs. for customer names

4

CURRENT SOLUTION

Host groups to the rescue!

CURRENT SOLUTION

Host groups to the rescue!

- Each device must belong to 3 host group categories
 - TECH-*
 - ALERT-*
 - CUST-*
- Optional host groups that override default behavior
 - AUTOTICKET-*
- Membership of these mandatory host groups can be monitored by Zabbix as well 😊
 - Eliminates unnoticed misconfigurations

CURRENT SOLUTION

Host groups to the rescue!

- Each device must belong to 3 host group categories
 - TECH-*
 - ALERT-*
 - CUST-*
- Optional host groups that override default behavior
 - AUTOTICKET-*
- Membership of these mandatory host groups can be monitored by Zabbix as well 😊
 - Eliminates unnoticed misconfigurations

CURRENT SOLUTION

Host groups to the rescue!

- Each device must belong to 3 host group categories
 - TECH-*
 - ALERT-*
 - CUST-*
- Optional host groups that override default behavior
 - AUTOTICKET-*
- Membership of these mandatory host groups can be monitored by Zabbix as well 😊
 - Eliminates unnoticed misconfigurations

CURRENT SOLUTION

TECH- host groups

- Examples: TECH-Networking, TECH-IT_Support, TECH-Datacenter, ...
- Determines who gets alerts
- 1-to-1 mapping with teams within the organisation

CURRENT SOLUTION

ALERT- host groups

- ALERT-24x7, ALERT-8x5, ALERT-NONE
- Determines when alerts are sent out
- ALERT-NONE is useful for:
 - Data collection to help troubleshooting
 - “Fake” hosts in Zabbix

CURRENT SOLUTION

CUST- host groups

- Examples: CUST-Skaro, CUST-Slitheen, ...
- Every customer has a host group
 - Nested host groups are used for customers with multiple sites
- Very convenient to schedule maintenance windows for planned outages
- We still follow the convention of prefixing hostnames with customer names

5

EXAMPLE ALERT ACTION

EXAMPLE

Conditions

Action Operations Recovery operations Update operations

* Name

Type of calculation Custom expression A and B and (C or D) and E and F

Label	Name	Action
A	Problem is not suppressed	Remove
B	Trigger severity is greater than or equals <i>High</i>	Remove
C	Host group equals <i>ALERT-8x5</i>	Remove
D	Host group equals <i>ALERT-24x7</i>	Remove
E	Host group equals <i>TECH-IT_Support</i>	Remove
F	Time period in <i>1-5,08:30-17:00</i>	Remove

Operations

Action Operations Recovery operations Update operations

* Default operation step duration

Default subject

Default message

Pause operations for suppressed problems

Operations	Steps	Details	Start in	Duration	Action
	2	Send message to user groups: 00000_IT TC via Email	00:05:00	Default	Edit Remove
	2	Send message to user groups: 00000_IT TC via Telegram	00:05:00	Default	Edit Remove
	4	Send message to user groups: 00006_IT Escalation via Email	00:15:00	Default	Edit Remove
	4	Send message to user groups: 00006_IT Escalation via Telegram	00:15:00	Default	Edit Remove
	7	Send message to user groups: 00006_IT Escalation 2nd via Telegram	00:30:00	Default	Edit Remove
	7	Send message to user groups: 00006_IT Escalation 2nd via Email	00:30:00	Default	Edit Remove

[New](#)

6

FUTURE DEVELOPMENTS

Always room for improvement...

FUTURE DEVELOPMENTS

Always room for improvement...

- SDMs want to see everything about specific customers
 - We can control access based on CUST- groups as well as TECH- groups
- Even more fine-grained control through tags?
 - Not researched yet
- Better monitoring of host configurations
 - Now: straight-on “one TECH-, one ALERT-, one CUST- group per host”
 - Sometimes it makes sense to e.g. have one host belong to multiple TECH- groups

FUTURE DEVELOPMENTS

Always room for improvement...

- SDMs want to see everything about specific customers
 - We can control access based on CUST- groups as well as TECH- groups
- Even more fine-grained control through tags?
 - Not researched yet
- Better monitoring of host configurations
 - Now: straight-on “one TECH-, one ALERT-, one CUST- group per host”
 - Sometimes it makes sense to e.g. have one host belong to multiple TECH- groups

FUTURE DEVELOPMENTS

Always room for improvement...

- SDMs want to see everything about specific customers
 - We can control access based on CUST- groups as well as TECH- groups
- Even more fine-grained control through tags?
 - Not researched yet
- Better monitoring of host configurations
 - Now: straight-on “one TECH-, one ALERT-, one CUST- group per host”
 - Sometimes it makes sense to e.g. have one host belong to multiple TECH- groups

7

CONCLUSION

CONCLUSION

- Using host groups as directives for logic works really well
 - Multiple dimensions are orthogonal
 - Can be set independently from each other
- Setting up initially is a hassle
 - A lot of very repetitive Trigger actions (one for every TECH- and ALERT- combo)
 - Once configured, adding new hosts to the system is easy
- Expanding functionality in the future should not be too difficult

CONCLUSION

- Using host groups as directives for logic works really well
 - Multiple dimensions are orthogonal
 - Can be set independently from each other
- Setting up initially is a hassle
 - A lot of very repetitive Trigger actions (one for every TECH- and ALERT- combo)
 - Once configured, adding new hosts to the system is easy
- Expanding functionality in the future should not be too difficult

CONCLUSION

- Using host groups as directives for logic works really well
 - Multiple dimensions are orthogonal
 - Can be set independently from each other
- Setting up initially is a hassle
 - A lot of very repetitive Trigger actions (one for every TECH- and ALERT- combo)
 - Once configured, adding new hosts to the system is easy
- Expanding functionality in the future should not be too difficult

**THANK
YOU!**

